

REMARKS

Reconsideration and allowance of all the claims of record is respectfully requested. Claims 11, 32, 51 and 72 have been cancelled hereby. Currently, claims 1-10, 12-21, 23-31, 33-50, 52-61, 63-71, and 73-85 are pending in the application.

Rejections Under 35 U.S.C. §103

Claims 1-8, 14-16, 18, 20-29, 35-37, 39-48, 54-56, 58, 60-69, 75-77, 80, and 81-85 stand rejected under 35 U.S.C. §103 as allegedly being unpatentable over Othmer et al (U.S. Pat. 6,167,358 hereinafter “Othmer”) in view of Wygodny et al (U.S. Pat. 6,282,701 hereinafter “Wygodny”). Applicants respectfully submit that the applied references fail to teach or suggest each of the elements of the claims. Thus Applicants respectfully request reconsideration and withdrawal of the rejections and allowance of the claims.

For example, independent claim 1, recites, *inter alia* “at least one bug tracking related menu, the contents of which vary based on the user's role in the software development process.” Independent claims 21 and 61 and dependent claim 42 also recite similar language. Independent claim 81 recites, *inter alia*, “at least a first bug tracking related menu, the contents of which vary based on the first user's role in the software development process; ... at least a second bug tracking related menu, different from the first bug tracking menu, the contents of which vary based on the second user's role in the software development process.”

The Office Action concedes Othmer does not teach “at least one bug tracking related menu, the contents of which vary based on the user's role in the software development process,” and to compensate for this deficiency, the Office Action introduces Wygodny. Addition of Wygodny, however, does not cure this defect.

According to the Office Action, Wygodny discloses this claim element when Wygodny states “developer uses a program called the BugTrapper analyzer 106 to create a file called a trace control information (TCI) file. The TCI file contains instructions that specify what information is to be collected from a program to be traced.” The Office Action continues, “it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method of bug tracking related menu tailored to the user's role in the software development process as taught by Wygodny...” Thus, it appears that the Office Action alleges that a TCI file that “contains instructions that specify what information is to be collected from a program to be traced” is, to one of skill in the art, no different from a menu tailored to a user's role in the software development process.” Applicants respectfully submit that this is a misinterpretation of Wygodny's teachings.

While the TCI file of Wygodny may be customizable, Applicants submit that a customized file is not a menu, nor is it customized based on a user's role. Simply because Wygodny allegedly teaches customizing a trace file based on a program to be traced, this does not mean that Wygodny teaches a menu, the contents of which vary based on a user's role in the software development process.

Wygodny teaches that the developer “generates the TCI file by using a trace options editor program.” (2:60-64; 5:26-30) Then, the developer “sends the TCI file for the particular program to a remote site.” (3:30-32; 5:37-41). The client cannot even view the TCI file, as it is encrypted (6:26-34; 6:39-41). Once the file has arrived, the client executes an agent file that attaches to the client program and runs the trace, based on the information in the TCI file. (5:65-6:5).

Nowhere, in the cited portions or the numerous other portions mentioning the TCI file, does Wygodny disclose or suggest that the TCI file is anything resembling a menu, the contents of which vary based on a user’s role in the software development process. The TCI file is presented as a file containing a certain set of instructions that tell a client-side trace library what data to collect. Applicants fail to see how this can be characterized as a menu, the contents of which vary based on a user’s role in the development process.

It is instructive to note that Wygodny uses the word “menu” eighteen times, and not once does it equate the TCI file to a menu. (8:55-60; 9:46-10:30; 11:19-20; 12:24-26; 15:37-38; 18:22-23; 20:38-41; 20:52-59; 21:14-17; 21:49-51). Even Wygodny itself distinguishes between a “menu” and a “customizable file.” Menus are used in the creation of the TCI file, but there is no variance taught in the menus of the Bug Trapper program. There is nothing in Wygodny to support the notion that a TCI file is, or could be construed as, a menu. It is a set of instructions, created by a developer, to instruct a trace. Nothing more.

Applicants have used the word “menu” as it is commonly understood by one of skill in the art (including, as shown above, as understood by Wygodny). Interpreting “menu” to equate to a file built by a developer in order to run a bug trace is unreasonable and has no foundation in common knowledge in the art, Applicants’ disclosure, or the teachings of Wygodny.

Applicants note that the Office Action states “As understood from the Applicants specification and drawings that menu is the information display to the user depending on their role. Wygodny explicitly discloses menu is displayed to user (in this case developer) to select the execution files.” From this, it would appear that the Examiner may be taking a position that the menus shown to the developer in Wygodny, and not the TCI file, teaches the menu. If that is the case, then Wygodny clearly does not teach a menu that varies based on a user’s role. The developer gets the same menu every time he or she logs on. There is no teaching of anyone having a different role logging into the Bug Tracker, so it is unreasonable to suggest that even the notion of a second menu, having different options based on a different role, exists. The only other “role” that could exist is that of a user, but not only is there no teaching that the user ever accesses the Bug Tracker program, Wygodny explicitly notes that the user cannot even see the contents of the TCI file. (6:26-34).

Even if Othmer or Wygodny does teach users having different roles, there is nothing to teach or suggest that menus would be varied based on those roles. Rather, it would be reasonable, absent such teaching, to assume that everyone gets the same set of

menus when logging into a system, as that is a conventional method for providing software menus.

The menu is more than just “information displayed to the user based on their role,” as the Office Action characterizes it. An example of a menu is a list of options available to a user, not simply any information displayed to the user. The menu is not simply a customized file. Even this Response is stored in a customized file (i.e., a customized word processing file), but it certainly is not a menu. Thus, menus that vary based on a user’s role would be, for example, different options displayed to different users having different roles. This is not shown in the cited portions of the applied references or elsewhere.

For at least these reasons, Applicants submit that claims 1, 21, 42, 61, and 81 are allowable. Claims 2-8, 14-16, 18, 20, 22-29, 35-37, 39-40, 42-48, 54-56, 58, 60, 62-69, 75-77, 80, and 82-85 should be allowable based at least on their dependency from allowable independent claims.

Applicants also respectfully submit that the cited combination of Othmer and Wygodny does not teach or suggest all limitations of claim 41 either. For example, claim 41 requires “processing user identification information including a password, wherein the processing includes determining the aspects of a system that a user is entitled to access based on a user's role in the development process.”

The Office Action alleges that Othmer teaches the underlined portion of this language, but Applicants submit that the cited “the server uses the user ID to associate

static information with a particular client machine and to keep a record about a particular user” in Othmer does not teach the claimed “determining the aspects of a system that a user is entitled to access based on a user's role in the development process.” Othmer does teach using a user ID, but fails to teach or suggest “determining the aspects of a system that a user is entitled to access based on a user's role in the development process.”

According to Othmer, “...the server generates and transmits a new user ID back to the client computer. The client computer then attaches this user ID to future black box information sent to the server. The server uses the user ID to associate static information with a particular client machine and keep a record about a particular user...The server stores the dynamic black box data ... tagged with the user ID...” (Col. 13, line 56 – Col. 14, line 13.) Nothing indicates that the server of Othmer “determin[es] the aspects of a system that a user is entitled to access based on a user's role in the development process” as a part of “processing user identification information.”

Further, although the Examiner has alleged that the portions of Othmer teaching the claimed features have been clearly pointed out, what appears to be missing is any explanation of how these portions allegedly teach the claimed features. Othmer teaches a plurality of users. Othmer keeps a record about a user. Othmer associates static information with a machine. However, Othmer does not teach or suggest “determining the aspects of a system that a user is entitled to access.” Othmer also does not teach or suggest such a determination made “based on a user's role in the development process.” The missing teachings are not obvious or apparent logical interpretations, leaps, or fill-

ins, and yet, absent any teaching of the noted claim language, and absent any reasoning as to how the noted prior art portions allegedly teach the claim language, the Office Action simply makes conclusory statements that the claimed features "are taught."

For at least this reason, Applicants submits that claim 41 is allowable over the prior art of record. Claims 42-60 should be allowable based at least on their dependency from allowable claim 41.

Claims 9-13, 30-34, 49-53 and 70-74 stand rejected under 35 U.S.C. §103(a) as allegedly being unpatentable over Othmer and Wygodny in view of Johndrew et al. (U.S. Publication no. 2001/0049697, hereinafter "Johndrew"). Claims 17, 38, 57 and 78 were further rejected under 35 U.S.C. §103(a) as being unpatentable over Othmer and Wygodny in view of admitted prior art (Applicants' specification, page 2, lines 9-11, hereinafter "prior art"). Claims 19, 40, 59 and 79 were also rejected under 35 U.S.C. §103(a) as allegedly being unpatentable over Othmer in view of Tse (U.S. Patent No. 5,742,754, hereinafter "Tse"). Neither Johndrew, the prior art, nor Tse cures the above noted deficiencies of the Othmer/Wygodny combination with respect to the base independent claims, however. Thus, based at least on their dependency from allowable independent claims, these claims should all be allowable.

Conclusion

In view of the foregoing, Applicants believe that all the claims are in condition for allowance, and an action to that end is earnestly solicited. If any issues remain to be

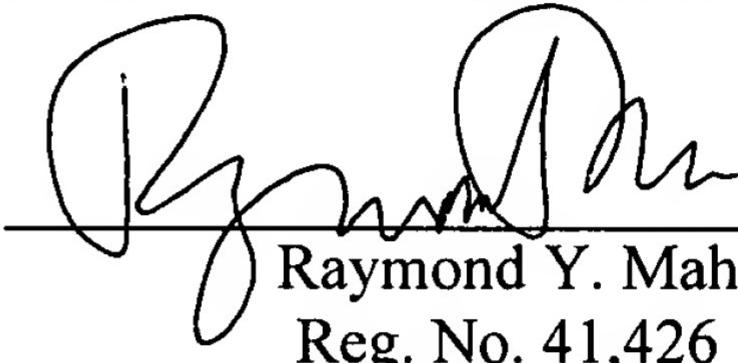
KELBAUGH et al.
Appl. No. 09/827,332
August 16, 2007

resolved, the Examiner is urged to contact Applicants' attorney at the telephone number identified below.

Respectfully submitted,

NIXON & VANDERHYE P.C.

By:



Raymond Y. Mah
Reg. No. 41,426

RYM:bpt
901 North Glebe Road, 11th Floor
Arlington, VA 22203-1808
Telephone: (703) 816-4000
Facsimile: (703) 816-4100